C^2: Truthful Incentive Mechanism for Multiple Cooperative Tasks in Mobile Cloud

Citation for published version:
Luo, S, Sun, Y, Wen, Z & Ji, Y 2016, C^2: Truthful Incentive Mechanism for Multiple Cooperative Tasks in Mobile Cloud. in *2016 IEEE International Conference on Communications (ICC).* Institute of Electrical and Electronics Engineers (IEEE), pp. 1-6, 2016 IEEE International Conference on Communications, Kuala Lumpur, Malaysia, 23/05/16. https://doi.org/10.1109/ICC.2016.7511052

Digital Object Identifier (DOI):
10.1109/ICC.2016.7511052

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
2016 IEEE International Conference on Communications (ICC)

C^2: Truthful Incentive Mechanism for Multiple Cooperative Tasks in Mobile Cloud

Shuyun Luo*, Yongmei Sun*, Zhenyu Wen†, Yuefeng Ji*
*the State Key Laboratory of Information Photonics and Optical Communications,
Beijing University of Posts and Telecommunications, Beijing, China
†Center for Intelligent Systems and their Applications, University of Edinburgh, UK
shuyunluo@gmail.com, ymsun@bupt.edu.cn, zwen@inf.ed.ac.uk, jyf@bupt.edu.cn

***Abstract*—In the practical crowdsourcing systems, there exist many cooperative tasks, each of which requires a group of users to perform together, such as finding the shortest multi-hop path or obtaining the media resources from a set of hosts. In this paper, we tackle the problem of how to truthfully and fairly schedule or allocate sufficient users who join mobile crowdsourcing applications with their smartphones. Moreover, the cooperation among users is taken into account. Thus, we present a novel Cooperative Crowdsourcing (C^2) auction mechanism for crowdsourcing multiple cooperative tasks. C^2 contains two parts: user selection and payment computation. In the first part, we first prove that users selection with the minimum social cost is NP hard problem and design a greedy algorithm to achieve near-optimal solution in polynomial time. The other part is that the server determines the payments of selected users to avoid the bidder's cheating behavior through a pricing algorithm that if and only if users honestly bid their cost, they can obtain the maximum utility. Both theoretical analysis and extensive simulations demonstrate that C^2 auction achieves not only truthfulness, individual rationality and high computational efficiency, but also low overpayment ratio.**

I. Introduction

Mobile crowdsourcing is the classic application of mobile cloud computing, which is the process of obtaining needed services or contents by soliciting contributions from a large group of mobile smartphone users. In the real world, there is increasing number of cooperative tasks, and this type of task needs a group of users to perform in terms of the requirements of applications. Crowd translator [1] recruits smartphone users from among native speakers of the target language. Since sufficient users have contributed for the same corpus, a corpus is created to train a speech recognizer. Doing a survey needs a large sample space such that its result has the statistical meanings. All above applications require users' collective contribution in order to complete those cooperative tasks.

User participation in the mobile crowdsourcing will incur resource cost, such as time, battery, bandwidth [2], which is constrained for personal devices. In general, any rational person will not provide the sensing or computing service voluntarily, unless an acceptable reward is offered to compensate the cost. Therefore, many incentive mechanisms have been proposed to motive users to contribute their resources. However, they either focus on the multiple independent task scenario [3]–[8], where each task only needs one user to perform, or pay attention to the single cooperative task scenario [9]–[11], where the task requires a group of users to perform cooperatively, which fails to consider interrelation among various tasks. [3] presents incentive mechanisms for both platform-centric and user-centric models. However, it assumes that the users' cost is a public knowledge. This is neither a practical in most mobile sensing system nor feasible for the cooperative crowdsourcing systems. Besides, the user-centric model only suits for simple tasks auction. The authors of [4] consider the cooperative task which needs to recruit enough users to collaborate, but it only concerns one task, and can not be directly used in multiple tasks case. In a word, it is vital to design a novel incentive mechanism for crowdsourcing system with multiple cooperative tasks.

In this paper, we introduce a procurement auction framework, in which the server announces the requirements of all candidate tasks, then users notice its task-bid information to the server, finally the server decides the task allocation policy and the payments for each user winner. Specifically, we design a novel metric, called cpv, to evaluate the cost per value obtainment. Based on the metric cpv, we propose a greedy algorithm to minimize the payment subject to the target of task value. If the payment for each user winner equals to its bid, the selfish behavior of each user results in the chaos of bidding, i.e., each user tends to bid a high price than his (her) real cost to get more payment. Based on the design rational of truthful mechanisms, we further propose a corresponding pricing algorithm to determine the payment of each user, whereby it can satisfy the property of truthfulness. Through rigid analysis of C^2, we demonstrate that C^2 not only achieves truthfulness, individual rationality and high computational efficiency, but also in low overpayment ratio.

With the goal of minimizing the server's payment, C^2 auction selects users and pays selected users with the constraint of task value target. Specifically, there are two contributions in this paper.

1) We first prove that users selection with the minimum social cost is NP hard problem and design a C^2 greedy algorithm to achieve near-optimal solution in polynomial time. Furthermore, the server determines the payments of selected users to avoid the bidder's cheating behavior through a pricing algorithm that if and only if users honestly bid their cost, they can obtain the maximum

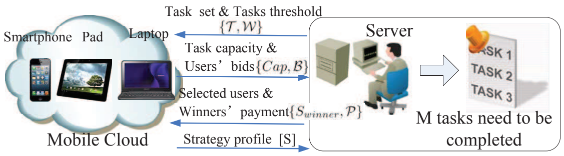


Fig. 1. A Cooperative Crowdsourcing System

utility.

2) To satisfy the property of truthfulness, we present a pricing algorithm to determine the payment of each selected user. Both theoretical analysis and extensive simulations demonstrate that C^2 auction achieves not only truthfulness, individual rationality and high computational efficiency, but also low overpayment ratio.

The rest of this paper is organized as follows. Section II presents the system model. In Section III, we present the detailed design of C^2 auction mechanism. Section IV evaluates the performance of our proposed mechanisms. Finally, Section V concludes the paper and looks forward to some possible future work.

II. System Model

In this section, we describe the crowdsourcing system model with multiple cooperative tasks, where the server recruits smartphone users in the mobile cloud to perform tasks collaboratively.

The system includes a server s and a set of users $\mathcal{U}$, $\mathcal{U} = \{1, \ldots, u_i, \ldots, u_N\}$. The server has a set of tasks needed to be completed $\mathcal{T} = \{1, \ldots, t_j, \ldots, t_M\}$. t_j requires at least m_{t_j} users to perform, and m_{t_j} is named as the *Task Threshold* for t_j. The set of Task Threshold is denoted as $\mathcal{W}$, i.e., $\mathcal{W} = \{m_{t_1}, \ldots, m_{t_j}, \ldots, m_{t_M}\}$. The sever announces task set $\mathcal{T}$ and the corresponding task thresholds $\mathcal{W}$ to users, and the value of t_j (noting as v_{t_j}) is obtained, if and only if it can recruit m_{t_j} users to perform.

Moreover, each user u_i has a capacity to perform a subset of tasks $SubT_{u_i}$, $SubT_{u_i} \subset \mathcal{T}$. The set of users' task capacity is denoted as Cap, i.e., $Cap = \{SubT_{u_1}, \ldots, SubT_{u_N}\}$. To do its selected tasks, u_i has an associated cost c_{u_i} for providing the resources, which is private and only known by itself. Thus, u_i claims the bid b_{u_i} for selling its service no less than its cost, i.e., $b_{u_i} \geq c_{u_i}$, and the set of all users' bids are denoted as $\mathcal{B}$.

The server interacts with the users through a four-step process, as shown in Fig. 1.

1) The server advertises task set $\mathcal{T}$ and the corresponding task thresholds $\mathcal{W}$ to users.
2) Each user announces the task-bid pair $(SubT_{u_i}, b_{u_i})$ to the server, where b_{u_i} is the reserved price u_i wants to sell its service.
3) Based on the task-bid pairs, the server selects a subset of users $\mathcal{S}_{winner} \subset \mathcal{U}$ as the winners and computes the payment $p_{u_i} \geq b_{u_i}$ for each winning user u_i, where $u_i \in \mathcal{S}_{winner}$. Then s announces the results to the users and pays the selected users with computed payment.
4) Each selected user u_i performs the tasks in its winning bid, and sends the sensed data back to the server.

Thus, the utility of each winning user u_i can be defined as:

$$Utility_{u_i} = p_{u_i} - c_{u_i} \tag{1}$$

The goal of the server is to select users and decide the payment such that the server's total payment is minimized meanwhile the server can obtain the required value V_{th}, which can be expressed as:

Objective: Minimize $\sum_{u_i \in \mathcal{S}_{winner}} p_{u_i}$

$$s.t. \sum_{t_j \in T_{comp}} v_{t_j} \geq V_{th} \tag{2}$$

where T_{comp} is the set of completed tasks.

C^2 auction mechanism determined by the server is assumed to be the common knowledge among the users. In the system, we also consider the existence of selfish users who attempt to bid the price to make their own utility as high as possible. Users are also assumed do not collude.

In this paper, we aim to design an incentive mechanism satisfying the following properties:

- Computational Efficiency: the solution can be computed in polynomial time.
- Individual Rationality (IR): each participating user will have non-negative utility.
- Incentive compatibility (IC): also called truthfulness, and each user always prefers reporting his private information truthfully to the server rather than any potential lie, i.e.; the user will get the maximum utility when the bided price is equal to its real cost.

III. Incentive mechanism Design

In this section, we first prove that the user selection problem is NP hard, thereby failing to exploit the well-known Vickrey-Clarke-Groves (VCG) mechanism to solve it. Furthermore, we propose an incentive mechanism based on procurement auction, named as Cooperative Crowdsourcing (C^2) auction, which aims to minimize the total payment of the server, subject to the given value target.

A. Problem Description

The objective is to design an incentive mechanism that selects users to minimize the server's payment under the condition that the server earns the targeted value. Therefore, designing such incentive mechanism is an optimization problem, which can be defined as follows.

Definition 1: User Selection (US) problem: Given a set of users $\mathcal{U}$, the server selects a subset of users $\mathcal{S}_{winner}$ as workers such that server's total payment is minimized, subject to a given value target.

The US problem can be formalized as the description of Eq. 2.

It is easy to get that the total payment of the server is minimized with $p_{u_i} = b_{u_i}$. Hence, the server's total payment becomes

$$\sum_{i \in \mathcal{S}_{winner}} b_{u_i} \tag{3}$$

Theorem 1: The US problem is an NP hard problem.

Proof: It is proved that the US problem can be reduced to Weighted Multiple Set Cover (WMSC) problem in polynomial time, which has already been proved to be NP-hard in [12].

WMSC INSTANCE: There are n subsets $\{C_1, C_2, \ldots, C_n\}$ of the base elements set $\mathcal{E} = \{e_1, e_2, \ldots, e_m\}$, and a positive integer k as well as a positive-integer-valued m-tuple $(p_1, p_2, \ldots, p_m)$. Question: Does there exist a subset $C' \subseteq C$ of size k, such that every element e_i is covered for at least p_i times.

The mapping instance of the US problem is established as follows. Let $\mathcal{T}$ be the task set mapping to $\mathcal{E}$, where there is a task $t_j \in T$ for each $e_j \in E$. Corresponding to each subset $C_i \in C$, user $u_i \in \mathcal{U}$ can do the task set $SubT_{u_i}$, which contains tasks mapping to the elements in C_i. If every element e_i is covered for p_i times, the mapping task t_j is done by multiple users with the size of p_j.

Hence, it is obvious that q is a solution of WMSC instance, if and only if it is a solution of the mapping one of US problem. Moreover, the reduction from WMSC instance to US instance ends in polynomial time. ■

Since the well-known VCG mechanism requires that the selected set of users is always the one with the lowest cost, which is impossible to compute in polynomial time because the US problem is NP-hard. To realize truthfulness of users' bids while minimizing the payment with a target value for the server, we propose a non-VCG auction mechanism based on the procurement auction, the details of which is illustrated in the following section.

Table I lists the frequently used notations.

TABLE I
NOTATION LIST

Notation	Description
$\mathcal{S}_{winner}$	user winners
$\mathcal{S}_{curr}$	current selected users
$\mathcal{S}_k$	selected users in the k-th iteration
T_{comp}	current completed tasks
T^k_{comp}	completed tasks in the k-th iteration
$SubU_{t_j}$	users able to perform task t_j
p_{u_i}	payment to user u_i
b_{u_i}	u_i's announced bid
v_{t_j}	value of task t_j
V	current obtained task vlaue
V_{th}	given value target
G_{t_j}	for task t_j, the candidate groups
G	for all tasks, the candidate groups
g	one candidate group

B. Cooperative Crowdsourcing (C^2) Auction

We propose a novel incentive mechanism C^2 based on the procurement auction, which aims to minimize the server's total payment, subject to a given value target. C^2 auction mechanism contains two periods: (1) user selection and (2) payment computation. For user selection period, the candidate users are selected by groups from the perspective of tasks. For payment computation period, each selected user is paid by the highest bid until it is still selected, when all other users' bids remain the same.

Algorithm 1 C^2 User Selection

Input: Users set ($\mathcal{U}$), tasks set ($\mathcal{T}$), users' bids ($\mathcal{B}$), task thresholds ($\mathcal{W}$), task capacity ($\mathcal{L}$) and task value target (V_{th}).
Output: Selected users ($\mathcal{S}_{winner}$) and social cost (C).

Initialization: $\mathcal{T}_{uncom} = \mathcal{T}$, $\mathcal{S}_{cuur} = \varnothing$, $k = 0$ and $V = 0$.
for all task $t_j \in \mathcal{T}_{uncom}$ **do**
  Based on m_{t_j} and $SubU_{t_j}$, compute the candidate groups G_{t_j}.
end for
$G = \bigcup G_{t_j}$, for all task.
while $V < V_{th}$ **do**
  Select the set $S_k = \arg\min cpv(g)$, where $g \in G$.
  $\mathcal{S}_{curr} = \mathcal{S}_{curr} \cup \mathcal{S}_k$, $G = G \setminus \mathcal{S}_k$
  $V = V + \sum_{j \in T^k_{comp} \cap \mathcal{T}_{uncom}} v_{t_j}$
  Delete T^k_{comp} from $\mathcal{T}_{uncom}$
  $k = k + 1$
end while
$\mathcal{S}_{winner} = \mathcal{S}_{curr}$
$C = \sum_{i \in \mathcal{S}_{winner}} b_{u_i}$

1) User Selection: First, we propose a C^2 User Selection greedy algorithm to solve the US problem, which is illustrated in Algorithm 1. The basic idea of C^2 User Selection algorithm is to pick out the most cost-efficient user groups by iterations which have small bids and can realize large task value, until the value target has been reached. Combing these two criteria into the single metric,

$$\frac{\sum_{i \in \mathcal{S}_k} b_{u_i}}{\sum_{j \in T^k_{comp}} v_{t_j}} \tag{4}$$

is regarded as the "cost per value", where T^k_{comp} means the completed tasks when the user group $\mathcal{S}_k$ is selected in the k-th iteration. By selecting the user group $\mathcal{S}_k$, the tasks T^k_{comp} are completed at the cost of $\sum_{u_i \in \mathcal{S}_k} b_{u_i}$, which is the total bids of users in $\mathcal{S}_k$. We maintain the set $\mathcal{S}_{curr}$ of the current selected users and the set $\mathcal{T}_{uncom}$ of the remaining uncompleted tasks in each iteration. The selected user set $\mathcal{S}_k$ in iteration k is supposed to minimize the marginal cost per value realization, defined as

$$cpv(\mathcal{S}_k) = \frac{\sum_{i \in \mathcal{S}_k \setminus \mathcal{S}_{curr}} b_{u_i}}{\sum_{j \in T^k_{comp} \cap \mathcal{T}_{uncom}} v_{t_j}}. \tag{5}$$

For task t_j, we extract the user set $SubU_{t_j}$ able to perform t_j from all users. Since the size of $SubU_{t_j}$ is largely smaller than the total user set, the search range is compacted a lot. Within $SubU_{t_j}$, all subset groups with the size of task threshold m_{t_j} are the candidate groups, denoted as G_{t_j}. The union of G_{t_j} for all tasks is the whole candidate groups G, which is the search range of user winners. The server selects the user set $\mathcal{S}_k$ with the minimum *cpv* from G in the k-th iteration.

In Algorithm 1, each *while* loop (Lines $2-3$) computes the candidate groups G_{t_j} which are the subset groups with the size of task threshold m_{t_j}. In each *while-loop* iteration, the group with minimum *cpv* will be selected. In other words, the selected group has the most cost-efficient bids that make the "greatest advance" to achieve the target value with small payment. k represents the iteration round. The *while-loop* shows that the process of user selection is terminated until the task value is achieved to the given value target.

2) Payment Computation: Combined with C^2 User Selection algorithm, we design the pricing algorithm for the incentive mechanism. We extend and adapt Theorem 2 to make truth-telling a weakly dominant strategy for each user, such that only users who bids honestly for its cost can gain the best payment.

Theorem 2: Based on the theorem in [13] [14], an auction mechanism is truthful if and only if:

1) The user selection algorithm is monotone: If user u_i wins the auction by bidding b_{u_i}, it also wins by bidding $b'_{u_i} \leq b_{u_i}$.
2) Given the user selection algorithm, there is a unique truthful mechanism associated with this selection algorithm. The pricing algorithm pays each winner the critical value: the highest bid the user could claim and still win under the condition of all other users' bids being fixed.

Algorithm 2 C^2 Payment Determination

Input: User winners ($\mathcal{S}_{winner}$), candidate groups (G) and users' bids ($\mathcal{B}$)
Output: Critical Payments ($\mathcal{P}$)

1: $p_{u_i} = 0$ for all users $u_i \in \mathcal{U}$, $\mathcal{T}_{uncom} = \mathcal{T}$, $\mathcal{S}_{curr} = \varnothing$ and $V = 0$.
2: **for all** user $u_i \in \mathcal{S}_{winner}$ **do**
3: $\quad k = 0$
4: $\quad$ **while** $V < V_{th}$ **do**
5: $\quad\quad$ Select the set $\mathcal{S}_k = \arg\min cpv(g)$, where $g \in G$.
6: $\quad\quad$ $G_{\setminus\{u_i\}} = \{G, G \cap \{u_i\} = \varnothing\}$
7: $\quad\quad$ Select the set $\mathcal{S}_{k\setminus\{u_i\}} = \arg\min cpv(g_{\setminus\{u_i\}})$, where $g_{\setminus\{u_i\}} \in G_{\setminus\{u_i\}}$.
8: $\quad\quad$ $\mathcal{S}_{curr} = \mathcal{S}_{curr} \cup \mathcal{S}_{k\setminus\{u_i\}}$
9: $\quad\quad$ $V = V + \sum_{t_j \in \mathcal{T}_{comp}^{k\setminus\{u_i\}} \cap \mathcal{T}_{uncom}} v_{t_j}$
10: $\quad\quad$ Delete $T_{comp}^{k\setminus\{u_i\}}$ from $\mathcal{T}_{uncom}$
11: $\quad\quad$ $p_{u_i} = \max\{cpv(\mathcal{S}_{k\setminus\{u_i\}}) \times \sum_{t_j \in \mathcal{T}_{comp}^{k} \cap \mathcal{T}_{uncom}} v_{t_j} - B(\mathcal{S}_k \setminus \mathcal{S}_{curr}) + b_{u_i}, p_{u_i}\}$
12: $\quad\quad$ $k = k + 1$
13: $\quad$ **end while**
14: **end for**
15: Return $\mathcal{P} = \{p_{u_i}, u_i \in \mathcal{S}_{winner}\}$

In Algorithm 2, the *for-loop* (Lines $2-14$) is to compute the critical bid for each winner $u_i \in \mathcal{S}_{winner}$. In each *while-loop*, it is aimed to calculate u_i's maximum bid that he can still be selected in this iteration. Given the current selected users $\mathcal{S}_{curr}$ and remaining tasks $\mathcal{T}_{uncom}$, we first select the set $\mathcal{S}_k$ and $\mathcal{S}_{k\setminus\{u_i\}}$ with the minimum *cpv* from the group set G and $G_{\setminus\{u_i\}}$, respectively (Lines 5-7), where $G_{\setminus\{u_i\}}$ is the candidate groups without u_i. The maximum bid in each iteration is the sum of u_i's bid and the marginal bid difference between $\mathcal{S}_k$ and $\mathcal{S}_{k\setminus\{u_i\}}$. Let d denote the last iteration in which the obtained task value V achieves the given target V_{th}. In the end, we set the maximum of these d bids among the *while* loops to the critical value p_{u_i}, which can promise u_i to be selected at least in one iteration.

C. Properties of C^2 auction mechanism

In this section, we present rigid theoretical analysis to demonstrate C^2 auction mechanism can achieve the desired properties of individual rationality, truthfulness and computational efficiency.

1) Individual Rationality : In the C^2 payment algorithm, Line 5 aims to find the subset $\mathcal{S}_k$ with u_i in the minimum *cpv*, while Line 7 tries to find the subset $\mathcal{S}_{k\setminus\{i\}}$ without u_i in the minimum *cpv*. We can obtain the inequality $cpv(\mathcal{S}_k) \leq cpv(\mathcal{S}_{k\setminus\{i\}})$, otherwise u_i will not be selected in the user selection period. Thus, $\frac{B(\mathcal{S}_k \setminus \mathcal{S}_{curr})}{\sum_{j \in T_k \cap \mathcal{T}_{uncom}} v_{t_j}} \leq cpv(\mathcal{S}_{k\setminus\{u_i\}})$ Therefore,

$$p_{u_i} = \max\{cpv(\mathcal{S}_{k\setminus\{i\}}) \times \sum_{t_j \in \mathcal{T}_{comp}^{k} \cap \mathcal{T}_{uncom}} v_{t_j} - B(\mathcal{S}_k \setminus \mathcal{S}_{curr}) + b_{u_i}, p_{u_i}\} \geq b_{u_i} \tag{6}$$

It shows that all users' utility is non-negative.

2) Truthfulness: The monotonicity of the user selection algorithm can be proved easily since u_i bidding a smaller value could increase the *cpv* value of the subset with user u_i. Thus, user u_i must win in the current or an earlier iteration.

Next, we demonstrate that p_{u_i} is the critical value for user u_i, i.e. bidding higher p_{u_i} could prevent u_i from winning the auction otherwise u_i must become a user winner. Suppose u_i is selected in the k-th iteration. On the one hand, if $b_{u_i} > p_{u_i}$, u_i could neither win in the k-th iteration nor in the following iterations because there exists another subset without u_i having smaller *cpv* value or $V \geq V_{th}$. On the other hand, if $b_{u_i} < p_{u_i}$, u_i must be selected in $k-$th iteration, because *cpv* value of the subset with u_i is reduced, i.e.,

$$\begin{aligned} cpv(\mathcal{S}_k) &= \frac{b_{u_i} + B(\mathcal{S}_k \setminus (\mathcal{S}_{curr} \cup \{u_i\}))}{\sum_{t_j \in \mathcal{T}_{comp}^{k} \cap \mathcal{T}_{uncom}} v_{t_j}} \\ &< \frac{p_{u_i} + B(\mathcal{S}_k \setminus (\mathcal{S}_{curr} \cup \{u_i\}))}{\sum_{t_j \in \mathcal{T}_{comp}^{k} \cap \mathcal{T}_{uncom}} v_{t_j}} \end{aligned} \tag{7}$$

3) Computational Efficiency: First, we compute the time complexity of user selection algorithm. Since the number of candidate groups G for all $t_j \in \mathcal{T}_{uncom}$ is at most $C_p^{\max(m_{t_j})} \cdot M$, computing the *cpv* value of each subset is in $O(C_p^{\max(m_{t_j})} \cdot M)$ time (Lines 2-4), where $\max(m_{t_j})$ is the maximum task threshold, $p = \max\{|SubU_{t_j}|\}$, $SubU_{t_j}$ is the users able to perform task t_j and $C_p^{\max(m_{t_j})}$ is the $\max(m_{t_j})$-combination of the set p. For line 7, finding the subset $\mathcal{S}_k$ with the minimum cpv is also at most in $O(C_p^{\max(m_{t_j})} \cdot M)$ time. It is obvious that $\max(m_{t_j}) < p \ll N$ in the real cooperative system. Since there are M tasks and each *while-loop* will contribute at least one task, the number of *while-loop* is at most M. Hence, the *while-loop* (Lines $6-12$) takes $O(C_p^{\max(m_{t_j})} \cdot M^2)$, and the User Selection algorithm runs in $O(C_p^{\max(m_{t_j})} \cdot M^2)$ time.

Next, we compute the running time of payment determination algorithm. In each round of finding of minimum cpv group (Lines 5 and 7), the process similar to Line 7 of Algorithm 1 is realized. Since by selecting each group at least one task can be completed and the maximum number of users in each group is $\max(m_{t_j})$, the maximum selected users is $M \cdot \max(m_{t_j})$. Therefore, the outside *for-loop* (Lines $2-14$) takes $O(\max(m_{t_j})M \cdot C_N^{\max(m_{t_j})} \cdot M^2) = O(\max(m_{t_j})C_p^{\max(m_{t_j})}M^3)$, which dominates the whole auction. It is obtained that the running time of C^2 auction mechanism is bounded by $O(\max(m_{t_j})C_p^{\max(m_{t_j})}M^3)$.

IV. Performance Evaluation

To evaluate the performance of C^2 auction mechanism, extensive simulations are done to evaluate the impact of the main parameters. The evaluation includes five types of performance metrics as follows:

1) Social cost (C): The total cost of selected users. In the user selection period, it is aimed to choose the users to make social cost minimized, subject to the value target.
2) Approximation ratio (R): This is the main metric demonstrating the performance of user selection algorithm. It illustrates how the proposed greedy algorithm approaches to the optimal solution (denoted by OPT). $R = \frac{C}{OPT}$ where C means the social cost.
3) Overpayment ratio [7]: It is computed as $\gamma = \frac{P-C}{C}$, where P denotes the total payment by the proposed truthful mechanism. Hence, the overpayment ratio characterizes the cost of the server making each user insisting on the truthfulness.
4) Utility of all users: The utility of all users is recorded to show the property of Individual Rationality.
5) Running time: The running time of C^2 auction is also recorded to evaluate its computational efficiency.

A. Simulation Setup

Let δ denote the average fraction of users who can perform each task, and δN is the average number of users who can perform each task. In a large system with many tasks, only a small portion of users can perform each task due to its resource limitation. For instance, a user in Boston is not able to perform the tasks that require location-based data from New York. Hence, δ is expected to be very small, e.g., $\delta \ll 1$.

The value of each task, the cost of each user and the task threshold of each task are uniformly distributed over $[5, 10]$, $[1, 5]$ and $[3, 5]$, respectively. All the simulations in this paper were run on a PC with 2.9GHZ CPU and 4GB memory. Each simulation is repeated for 100 times, and the average values are reported as statistical results.

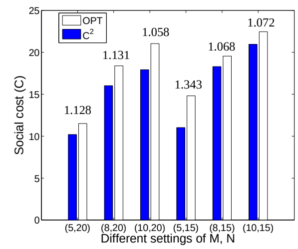


Fig. 2. Evaluation of approximate ratio

B. Evaluation of Approximation Ratio

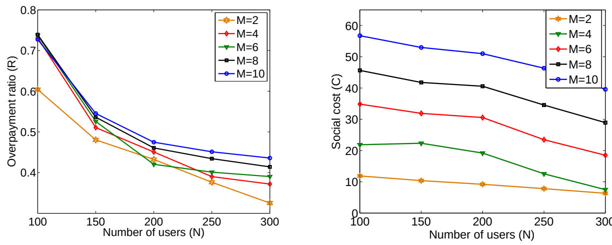
We first evaluate the performance of C^2 user selection algorithm contained in our proposed auction. Since US problem is NP-hard, it is time consuming to obtain the optimal solution (noted as OPT) with the general approach, i.e., brute force search. Hence, the approximate ratio of C^2 are only be evaluated in the settings with small scale. Specifically, the total number of users N is chose as 15 and 20, while the number of tasks M is picked as 5, 8 and 10. To reduce the execution time, the users' groups are iterated by starting from the minimum task threshold and terminating when the value target is reached. Moreover, we set $\delta = 0.2$ to define the number of users involving the selection, and the target value V_{th} is set as the total task value minus 5, i.e., $V_{th} = V_{total} - 5$.

In Fig. 2, we show the approximate ratios of C^2 in various settings, which are the numbers located over bars. It is clear that the social costs of C^2 user selection method under all listed settings are very close to that of optimal solutions. Compared with the cases of $N = 15$ and $N = 20$, the social cost of C^2 has declining trend when $N = 20$. The reason is that the augment of users resource can make the server have better choices. With the augment of M, it is shown that the social cost increases dramatically. That is because the server needs to recruit more users to complete tasks.

C. Evaluation of Overpayment Ratio

We evaluate the impact of the number of users (N) on overpayment ratio. N is varied from 100 to 300 with the increment of 50, and M from 2 to 10 with the increment of 2. It is set that $\delta = 0.1$ and $V_{th} = V_{total} - 10$. As can be seen in Fig. 3 (a), the overpayment ratio of C^2 auction keeps below 1 under different M and N, indicating C^2 auction with low overpayment cost for the truthful property. With the increase of N, the overpayment ratio decreases. The reason is that, since the augment of candidate groups, the cost bridge between the minimum candidate group and the second minimum one is suppressed. In addition, with the increase of M, the overpayment ratio rises accordingly. That is because the number of selected users increase with the requirement of more tasks completion.

Fig. 3 (b) shows that the social cost decreases with the rising number of users and has the opposite trend with the increasing number of tasks.



(a) Impact on overpayment ratio (b) Impact on social cost

Fig. 3. The impact of N and M on overpayment ratio and social cost

D. Evaluation of Individual rationality

In order to show all users have non-negative utility, we depict the empirical CDF of the utility for all users under various settings. From Fig. 4, it is observed that the proportion of users with negative utility is zero. Only the selected users have positive utility, thus most of users have zero utility, which can be shown when utility equals to zero in Fig. 4. Hence, it is confirmed that all users have non-negative utility, which illustrates that C^2 auction mechanism achieves the property of individual rationality.

E. Evaluation of Computational Efficiency

Fig. 5 demonstrates the computational efficiency of C^2 auction with different settings, which shows the running time of all cases is under 10 seconds. Based on the research results from the response time in man-computer conversational transactions [15], 10 seconds is the limit for users keeping their attention on the task. Therefore, C^2 auction mechanism has high computational efficiency in the small scale.

V. Conclusion

In this paper, we conducted deep and complete incentive study on the crowdsourcing system with multiple collaborative tasks. Taking the correlation among tasks and users, we proposed a truthful incentive mechanism, cooperative crowdsourcing (C^2) auction to stimulate sufficient users for task completion, which is composed of a near-optimal approximate algorithm and a critical payment scheme. Through both theoretical analysis and extensive simulations, it is demonstrated that C^2 auction achieves not only truthfulness, individual rationality and high computational efficiency in small scale, but also low overpayment ratio. In the future, we shall extend this work into the online scenario for the real time applications, as well as the impact of users' mobility.

Acknowledgment

This research was supported by major Program of National Natural Science Foundation of China (No. 61190114).

References

[1] J. Ledlie, B. Odero, E. Minkov, I. Kiss, and J. Polifroni, "Crowd translator: on building localized speech recognizers through micropayments," *ACM SIGOPS Operating Systems Review*, vol. 43, no. 4, pp. 84–89, 2010.

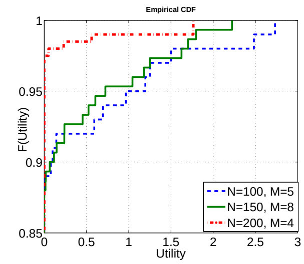


Fig. 4. Empirical CDF of utilities for all users

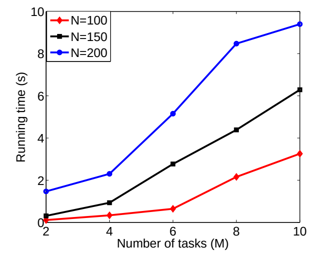


Fig. 5. Evaluation of running time

[2] G. Zhang, E. Ding, K. Yang, and P. Liu, "A suboptimal joint bandwidth and power allocation for cooperative relay networks: a cooperative game theoretic approach," *Science China Information Sciences*, vol. 56, no. 7, pp. 1–12, 2013.

[3] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 173–184.

[4] L. Duan, T. Kubo, K. Sugiyama, J. Huang, T. Hasegawa, and J. Walrand, "Incentive mechanisms for smartphone collaboration in data acquisition and distributed computing," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 1701–1709.

[5] L. G. Jaimes, I. Vergara-Laurens, and M. A. Labrador, "A location-based incentive mechanism for participatory sensing systems with budget constraints," in *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*. IEEE, 2012, pp. 103–108.

[6] Y. Zhang and M. van der Schaar, "Reputation-based incentive protocols in crowdsourcing applications," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 2140–2148.

[7] Z. Feng, Y. Zhu, Q. Zhang, M. L. Ni, and A. V. Vasilakos, "Trac: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014.

[8] T. Luo, H.-P. Tan, and L. Xia, "Profit-maximizing incentive for participatory sensing," in *IEEE INFOCOM*, 2014.

[9] Y. Lee, Y. Ju, C. Min, S. Kang, I. Hwang, and J. Song, "Comon: cooperative ambience monitoring platform with continuity and benefit awareness," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 43–56.

[10] A. Archer and É. Tardos, "Frugal path mechanisms," *ACM Transactions on Algorithms (TALG)*, vol. 3, no. 1, p. 3, 2007.

[11] X. Liu, K. Ota, A. Liu, and Z. Chen, "An incentive game based evolutionary model for crowd sensing networks," *Peer-to-Peer Networking and Applications*, pp. 1–20, 2015.

[12] J. Yang and J. Y.-T. Leung, "A generalization of the weighted set covering problem," *Naval Research Logistics (NRL)*, vol. 52, no. 2, pp. 142–149, 2005.

[13] N. Nisan, *Algorithmic game theory*. Cambridge University Press, 2007.

[14] Y. Singer, "Budget feasible mechanisms," in *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*. IEEE, 2010, pp. 765–774.

[15] "Response times: The 3 important limits," http://www.nngroup.com/articles/response-times-3-important-limits/.